

Peer Review

Review of: "In RAG We Trust? Measuring Robustness of Retrieval-Augmented Generation Under Document Poisoning"

Evgenii Arsentev¹

1. Independent researcher

1. What the work claims

The paper reports a factorial measurement of how a small, heavily quantized RAG stack behaves when the passages placed in its prompt are corrupted. The pipeline is fixed throughout: an **all-MiniLM-L6-v2** encoder, a **FAISS flat inner-product** index over 19,597 FEVER dev evidence sentences plus ten passages written by the author, **k = 3**, and a **Llama 3.1 8B Instruct quantized to 4 bits (GGUF Q4_K_M)** decoding at temperature 0.1 on CPU (Section II.A, Section III.A). Three rule-based corruption strategies (entity swap, number swap, negation; Section II.B) are crossed with four poisoning levels (0/3, 1/3, 2/3, 3/3; Section II.C) over 49 queries, giving $49 \times 3 \times 4 = 588$ runs and 1,176 generations (Section II.E). The headline claims are: accuracy falls from 77.9% clean to 43.5% at full corruption (Abstract; Table I, rows 0/3–3/3, columns "Clean acc." and "Poisoned acc."); entity swap flips the most answers (Table III, row "Entity swap", column "Fooled rate", 23.5%); numeric corruption is flat until poisoned passages form a majority and then jumps (Section III.E, 10.2% at 0/3 and 1/3 to 26.5% at 2/3 and 34.7% at 3/3); and the model's dominant reaction to contradiction is abstention rather than fabrication, with a lexical-overlap proxy of unsupported generation falling under attack rather than rising (Section III.F, Figure 4(b), roughly 29% clean versus 20% poisoned). The author is candid that this is a small-scale measurement with coarse automated labels and frames the strategy contrasts as suggestive (Abstract, final sentence; Section V).

2. Whether the conclusion holds on the data

Internal arithmetic: it reconciles, and this deserves to be said first. I recomputed every cell of Tables I and III as integer counts and cross-summed the two tables, which partition the same 588 runs along orthogonal axes. Table I (147 runs per row) implies 115, 114, 115, 114 clean-correct runs, and Table III (196

runs per row) implies 153, 152, 153: both sum to 458. Poisoned-correct: $102 + 94 + 78 + 64 = 338$ and $107 + 113 + 118 = 338$. Fooled: $14 + 22 + 38 + 50 = 124$ and $46 + 40 + 38 = 124$. Abstentions: $31 + 43 + 57 + 76 = 207$ and $85 + 59 + 63 = 207$. Column means also agree (Table I clean mean 77.9%, Table III 77.9%; fooled 21.1% both; abstention 35.2% both, matching the 35.2% quoted in Section III.F). The number-swap cells of Section III.E likewise resolve to 5, 5, 13, 17 out of 49 and average to the 20.4% in Table III, row "Number swap". The reported 76.9% clean/poisoned answer agreement at 0/3 (Section III.B) resolves to 113/147. I found **no arithmetic discrepancy**; the tables were evidently generated from one run log, as the paper states. The problems below are therefore problems of construct, design, and inference, not of bookkeeping.

The manipulation is not document poisoning, and the title and threat model do not match the mechanism. Section II.B states plainly that "the rewrite happens after retrieval and before the prompt is built, so the retriever itself is never attacked." The attack is therefore not calibrated to the retriever in any sense: there is no question of whether a poisoned document wins retrieval, no similarity budget, no measurement of whether the corrupted text would still have been returned in the top 3 by the MiniLM encoder. The effective poison rate over the stated corpus of $19,597 + 10 = 19,607$ passages (Section III.A) is **zero** – not one indexed document is modified. What is measured is prompt-context corruption with guaranteed delivery, i.e., the strongest possible attacker, so all numbers in Tables I-III are upper bounds on any pre-retrieval poisoning scenario. This matters because the threat model in Section IV names exactly the pre-retrieval scenarios it does not model: "a compromised document store, a poisoned web source that a live retriever pulls in." In both, the attacker must first get the poisoned text retrieved, which is the hard part and is assumed away. Either the title and Section IV should be narrowed to post-retrieval context tampering (a real and defensible threat, e.g., the "tampered cache" case, which is the one scenario of the three that the design does model), or documents must be poisoned in the index and retrieval success reported as its own metric.

The threat model appears too late to constrain the design. For a security paper, this is a first-order defect. The attacker model is stated only in **Section IV**, after every result; it belongs before Section II.B, since it determines what a valid attack implementation is. As stated, it is also incomplete on the dimensions a reader needs: it does not say how many documents the attacker may write or edit, whether the attacker knows the query in advance (the design implies full knowledge, since the corruption targets the already-retrieved evidence for that specific question), whether the attacker has a detectability budget,

or why an adversary powerful enough to edit text in transit cannot equally edit the index. No attacker cost model is given.

How poisoned passages are generated is never specified, and "poisoned" may not mean modified. The corruptions are rule-based, not LLM-generated (Section II.B gives three hand-worked examples; Section V calls them "rule-based edits" and proposes replacing them with model-generated adversarial passages as future work). But Section II.B gives no implementation: no NER model for locating entities, no candidate pool for "plausible alternatives of the same type," no perturbation distribution for numbers (the worked example "over 13,000 miles" to "over 3,000 miles" is a leading-digit deletion, a factor of 4.3, not a "nearby wrong" value), no parser or rule for negation insertion. More seriously, these rules can silently no-op: a FEVER evidence sentence containing no numeral cannot be number-swapped, and a passage whose main clause resists negation insertion is returned unchanged. The paper never reports the **edit-success rate per cell**, i.e., how many of the nominally poisoned passages were actually altered. This is not a detail – it is a complete alternative explanation for the paper's most emphasised finding. If number swap frequently fails to alter the single targeted passage at level 1/3, then the flatness from 10.2% to 10.2% (Section III.E) measures the failure rate of the rule, not the model's resistance to a minority of wrong figures, and the "majority-leaning pattern" (Section III.E, Section IV) dissolves.

The headline drop compares a column mean with a single cell, against the paper's own instruction. Section III.B is admirably explicit: "Attack effects should be read against the 0/3 residual, not against a silent zero." The Abstract and Section V then do exactly the opposite, reporting a fall "from 77.9% on clean context to 43.5%" – 77.9% being the mean of the clean column across all four rows of Table I, and 43.5% the poisoned cell of row 3/3. The like-for-like comparison within the poisoned arm is 69.4% (Table I, row 0/3) to 43.5% (row 3/3), a fall of **25.9 points, not 34.4**. The Abstract should carry the smaller, defensible number.

The fooled rate is not the quantity the text says it is. Section II.D defines it as isolating correct-to-incorrect flips "from questions the model could not answer anyway," but the denominator is clearly all runs, not clean-correct runs: at row 3/3 of Table I, 34.0% of 147 is 50 runs, exactly the 114 clean-correct minus 64 poisoned-correct. Conditioned on the answers the model actually knew, the flip rate at 3/3 is $50/114 = 43.9\%$, not 34.0%. Since the stated purpose of the metric is to condition on clean-correct, the conditional rate should be the one reported, in Tables I, II, and III alike. (Incidentally, the same reconstruction shows the number of clean-wrong-to-poisoned-correct runs is 1, 2, 1, and 0 across rows 0/3 to 3/3 of Table I – worth reporting, as it bounds the label noise.)

Recomputation reverses the paper's central ranking. At Table I row 3/3, poisoned accuracy is 43.5% and abstention is 51.7%; since an abstention cannot contain the reference keyword, correct and abstaining runs are disjoint and at most $100 - 43.5 - 51.7 = 4.8\%$ of runs (7 of 147) can be answers that are both non-abstaining and wrong. In other words, of the 50 "fooled" runs at full poisoning, at least 43 are refusals. The fooled rate is therefore largely a renaming of the abstention rate, and the security-relevant quantity – how often the model states a falsehood – is never reported. Applying the same subtraction to Table III: entity swap $100 - 54.6 - 43.4 = 2.0\%$ (4 of 196 runs), negation $100 - 60.2 - 32.1 = 7.7\%$ (15 of 196), number swap $100 - 57.7 - 30.1 = 12.2\%$ (24 of 196). By the measure a practitioner cares about, number swap is roughly six times more dangerous than entity swap, which is the exact opposite of the ranking in Table III's "Fooled rate" column (entity 23.5% > number 20.4% > negation 19.4%) and of the Abstract's claim that "entity swap flips the largest share of answers that were correct on clean context." Both statements are technically true of the metric as defined, and both mislead. The paper's own Section IV conclusion that "numeric facts deserve special protection" is in fact much better supported than Table III suggests. I flag that this recomputation rests on the assumption that abstaining answers never contain the reference keyword; the authors should confirm it against the run log and report the poison-adoption rate directly.

The significance test flags the null condition. Section III.E reports exact McNemar $p < 10^{-3}$ at 2/3 and 3/3 and $p = 0.22$ at 1/3. Applying the same test to the no-attack cell of Table I, row 0/3, using the counts reconstructed above (14 discordant pairs correct-to-wrong, 1 wrong-to-correct), gives an exact two-sided $p = 2 \times (1 + 15)/2^{15} = 9.8 \times 10^{-4}$, which is also below 10^{-3} . A test that declares the untreated control significant at the same threshold as the treatment is measuring decoding noise, not the attack. This does not mean the effects at 2/3 and 3/3 are unreal – their effect sizes are far larger – but it does mean the reported p-values carry no evidential weight as presented. The test must either be run against the 0/3 poisoned arm as the reference (not against the clean arm), or the pair must be made comparable by locking the decoding seed, as Section V already proposes.

The uncertainty analysis omits the dominant noise source and clusters at the wrong level. Section III.C resamples "the 49 indexed query slots with replacement (37 distinct texts)." Two problems follow. First, the resampling unit should be the 37 distinct queries, not the 49 slots: the 12 duplicated texts are not independent draws, so the intervals in Table II are anti-conservative. Second, and more importantly, the bootstrap resamples queries only – it cannot capture generation variability, because each cell contains exactly one generation pair per query and there are no repeated draws anywhere in the 588-run table. Yet Section III.B establishes that generation variability is the largest single nuisance in the study (clean and

poisoned answers agree in only 76.9% of the 0/3 runs). The intervals in Table II are thus uncertainty estimates that exclude the known dominant noise term. Relatedly, the 0/3 cells of the three strategies are three replicates of one and the same condition – 98 of the 588 runs are duplicated null cells – and their spread ($5/49 = 10.2\%$ for number swap against $9/98 = 9.2\%$ for the other two) is a free, unused estimate of exactly that noise. Finally, the interval for the number-swap jump, 16.3 points with CI [2.0, 30.6] (Section III.E), has a lower bound essentially at zero and is one of at least nine adjacent-level contrasts (3 strategies x 3 steps) reported without any multiplicity adjustment; under even Bonferroni correction it would not exclude zero.

The correctness metric confounds the strategy ranking it is used to establish. Section II.D defines accuracy as case-insensitive presence of a reference keyword and honestly warns that "a negated or qualified answer can still contain the keyword." That warning applies with full force to precisely one of the three strategies. If the poisoned passage says "Barack Obama was never the 44th President" and the model repeats it, the answer contains the keyword and is scored **correct**. The negation arm is therefore systematically credited for successful attacks, which is the simplest explanation for negation showing the lowest fooled rate (19.4%) and the highest poisoned accuracy (60.2%) in Table III, and for the claim in Section III.F and Figure 5 that "negation is the most resilient." That conclusion cannot be drawn from a substring matcher. It should be withdrawn or re-established on a polarity-aware label.

Attack success is not judged by an LLM – which avoids one bias and introduces another. To the paper's credit, there is no LLM-as-judge anywhere in the evaluation: accuracy, abstention, and the unsupported-generation proxy are all rule-based (Section II.D), so there is no judge-model bias or self-preference to argue about. The cost is that all three labels are brittle. Abstention is detected by "a small set of phrases" that is never enumerated; the phrase list determines the abstention column of Tables I and III and, through it, every recomputation above. The unsupported-generation proxy of Eq. 3 has three unspecified free parameters (the content-word/stopword definition, the 0.3 threshold, and – critically – whether W_d is the clean or the poisoned passage set). That last ambiguity determines the sign of the result in Figure 4(b): if W_d is the poisoned set, an answer that faithfully repeats the poison has high overlap and is not flagged, so the proxy is guaranteed to fall exactly when the attack succeeds. Section II.D concedes that the proxy "can miss answers that copy poisoned text," yet Figure 4(b) and the Abstract still present the fall from roughly 29% to 20% as a finding about unsupported generation. Given the mechanism, the fall is close to an artifact of the definition and should be reported as such or dropped.

Defenses are discussed but never evaluated. Section IV proposes three mitigations – cross-checking numbers against an independent source, flagging cross-passage disagreement, and monitoring the abstention rate as an early-warning signal – and Section V defers all of them to future work. Not one is implemented, so the paper offers no baseline against which any future defense can be measured, and the practical recommendations in Section IV are unsupported by any experiment in the paper. The claim that “raising k without raising source diversity offers little real protection” (Section IV) is particularly exposed: k is fixed at 3 in every one of the 588 runs (Section II.A, Section III.A), so no evidence about varying k exists anywhere in the study.

Generalization is untested on every axis. One generator, one quantization, one retriever, one index type, one k , one temperature, one corpus. Nothing supports extrapolation beyond this configuration, and the paper mostly says so (Section V). Two specific confounds should be named, however. First, the study attributes behavior to “a small quantized model” (Abstract, Section V) but cannot separate small from quantized without a full-precision Llama 3.1 8B control run; 4-bit Q4_K_M quantization is a plausible independent driver of the very high abstention rates (up to 69% for entity swap at full corruption, Section III.F). Second, the ten author-written “information-rich passages” (Section II.E) are the ones that actually answer the queries, and Section III.G reports that the most fragile questions are exactly those served by them “because those passages contain the richest set of swappable entities.” The magnitude of the headline effect is therefore a property of ten unpublished passages the author wrote, not of FEVER.

Positioning. Section III.H compares the work only to non-adversarial RAG papers ([1], [6]), a distraction study ([2]), and a benchmark-contamination note ([7]). No prior work on retrieval-corpus poisoning, knowledge-base corruption, or prompt injection is cited or contrasted anywhere in the seven references, so a reader cannot determine what in this paper is new. For a paper whose contribution is framed as a security measurement, the adversarial-RAG literature cannot be absent from related work.

3. Reproducibility

No code or data availability statement appears anywhere in the article. There is no repository link, no archived artifact, no DOI for data. Given that the study’s entire contribution is a 588-run measurement, this alone blocks independent verification.

Models. The generator is named as Llama 3.1 8B Instruct in GGUF Q4_K_M (Section II.A), but without the source repository or file, the quantizer, a file checksum, or the llama.cpp version or commit – all of which change outputs. Decoding is specified only as temperature 0.1; top_p, top_k, min_p, repeat penalty, max

tokens, context length (n_{ctx}), and the random seed are all unstated, and Section III.B confirms the generation seed is not fixed. The retriever is named (all-MiniLM-L6-v2, ref [3]) but without a model revision or sentence-transformers version; the FAISS index type is given as flat inner-product with L2-normalized vectors (Section II.A, Eq. 1), which is adequate, but no FAISS version is given. The prompt is described in prose ("answer only from the supplied context", reply "Not enough information", Section II.A) but never reproduced verbatim – for a paper about instruction-following under conflicting evidence, the exact system and user templates are load-bearing and must be printed in full.

Retrieval and chunking. $k = 3$ is stated (Section II.A). Chunking is never specified: the corpus is described as 19,597 FEVER dev evidence sentences (Section III.A), which implies sentence granularity, but the ten added passages are called "information-rich" and "longer" by implication (Section II.E) with no length, no chunk size, no overlap, and no statement of whether they were split before indexing. Which of the three retrieved passages is corrupted is chosen "at random among the three" (Section II.C) with no seed given and no statement of whether the choice is matched across strategies; if it is not matched, position effects are confounded with strategy in Table III.

Corpus and queries. The FEVER dev subset is named (ref [5]) but the split, version, and filtering procedure are not. FEVER is a claim-verification set with SUPPORTED/REFUTED/NOT ENOUGH INFO labels, while this study uses free-form questions with "reference keywords" (Section II.D); the conversion procedure from FEVER claims to the 49 queries is never described, which is a substantial undocumented step. Neither the 49 queries (nor the 37 distinct texts), nor their reference keywords, nor the ten added passages, nor the abstention phrase list, nor the stopword list for Eq. 3 are provided. Why 12 of the 49 queries are duplicates is not explained (Section II.E). The poisoning rules themselves have no pseudocode, no regular expressions, no entity type inventory, and no worked failure cases beyond the three illustrative examples in Section II.B.

Seeds. The bootstrap seed is described as fixed (Section III.C) but its value is not given; the number of resamples (10,000) is. The generation seed is explicitly not controlled (Section III.B, Section V), which the paper correctly identifies as a limitation. In its present form, a third party given the paper alone could not reproduce a single number in Tables I, II, or III.

4. What specifically to fix

1. **Move the threat model from Section IV to the head of Section II**, before the poisoning strategies, and complete it: how many documents the attacker may write or edit, whether the query is known

in advance (the design implies yes), the detectability budget, and why an attacker who can rewrite text in transit cannot rewrite the index.

2. **Reconcile the title and Section IV with the mechanism in Section II.B.** Since the rewrite is post-retrieval and the effective corpus poison rate over the 19,607 indexed passages (Section III.A) is zero, either retitle to post-retrieval context corruption and keep only the "tampered cache" scenario of Section IV, or poison documents in the index and add a retrieval-success rate as a first-class metric alongside Table I.
3. **Specify and release the corruption rules** (Section II.B): entity recognizer, candidate substitution pool, numeric perturbation distribution, negation insertion rule, with pseudocode.
4. **Report the edit-success rate per cell of Figure 2 and Section III.E** – what fraction of nominally poisoned passages were actually modified. Until this is reported, the flatness of number swap at 10.2% across levels 0/3 and 1/3 cannot be distinguished from the number-swap rule no-opping on passages that contain no numeral, and the majority-effect claim in Section III.E and Section IV is unsupported.
5. **Change the Abstract and Section V headline** from "77.9% to 43.5%" to the within-arm comparison 69.4% to 43.5% (Table I, rows 0/3 and 3/3, column "Poisoned acc."), i.e., 25.9 points, in line with the instruction the paper itself gives in Section III.B.
6. **Report the fooled rate conditional on clean-correct**, as Section II.D says the metric is meant to be. At Table I row 3/3 this is $50/114 = 43.9\%$, not 34.0%. Update Tables I, II, and III accordingly, and state the denominator explicitly in the caption of each.
7. **Add a poison-adoption rate** – the fraction of runs in which the answer asserts the injected falsehood – as a separate column in Tables I and III. At present, it is absent, and it is the only number a security reader needs. From Table I row 3/3, 43.5% correct plus 51.7% abstaining leaves at most 4.8% of runs (7 of 147) that can be non-abstaining and wrong; from Table III, the same subtraction gives at most 2.0% for entity swap, 7.7% for negation, and 12.2% for number swap. Please verify these against the run log.
8. **Revise the strategy ranking in Table III and in the Abstract.** By the non-abstaining-wrong measure derived in item 7, number swap (12.2%) is about six times more harmful than entity swap (2.0%), reversing the "Fooled rate" ordering of Table III (entity 23.5% > number 20.4% > negation 19.4%). The Abstract's "entity swap flips the largest share" should be qualified as an abstention-driven effect.

9. **Withdraw or re-establish the claim that negation is the most resilient strategy** (Section III.F, Figure 5, Table III row "Negation"). The keyword matcher of Section II.D scores a repeated negated falsehood as correct whenever it retains the keyword, so the metric systematically credits the negation arm for successful attacks. A polarity-aware or NLI-based label is required before any negation ranking can stand.
10. **Report an exact McNemar test on the 0/3 control cell of Table I as a negative control.** On the reconstructed counts (14 correct-to-wrong, 1 wrong-to-correct) it gives $p = 9.8 \times 10^{-4}$, below the same 10^{-3} threshold used in Section III.E to certify the attack at 2/3 and 3/3. Either lock the decoding seed across the clean/poisoned pair or test the poisoned arm against the 0/3 poisoned cell rather than against the clean arm.
11. **Resample at the level of the 37 distinct queries, not the 49 slots** (Section III.C, Table II), and state that the intervals exclude generation variability – or, preferably, add repeated generations per cell so that variability enters the interval. As Section III.B shows (76.9% pair agreement at 0/3), it is the largest noise term in the study.
12. **Correct for multiplicity on the adjacent-level contrasts of Section III.E.** The 16.3-point number-swap jump with CI [2.0, 30.6] is one of at least nine such comparisons, and its lower bound is 2.0; report the adjusted interval or drop the claim that it "excludes zero."
13. **State in Eq. 3 whether W_d is the clean or the poisoned passage set**, and give the content-word definition and a sensitivity analysis over the 0.3 threshold. If W_d is the poisoned set, the fall in Figure 4(b) from roughly 29% to 20% is a mechanical consequence of poison adoption raising overlap, and the result should be reported as an artifact rather than as evidence that unsupported generation declines under attack (Abstract, Section III.F).
14. **Enumerate the abstention phrase list** used in Section II.D. Every abstention figure in Tables I and III, and every recomputation that depends on them, is a function of that list.
15. **Print the full prompt template verbatim** (system and user), currently only paraphrased in Section II.A, and give the complete decoding configuration: `top_p`, `top_k`, `min_p`, repeat penalty, max tokens, `n_ctx`, and seed.
16. **Pin every artifact version** in Section II.A and Section III.A: the GGUF file and its checksum, the llama.cpp commit, the MiniLM model revision, sentence-transformers and FAISS versions, and the FEVER split and filtering procedure.
17. **Release the ten added passages, the 49 queries with reference keywords, the run log, and the analysis code**, and add a data and code availability statement – currently absent from the paper

entirely. Section III.G attributes the most fragile queries to these ten passages, so the central result is presently not inspectable.

18. **Specify chunking** (Section II.A, Section II.E): whether the ten added passages were split, with what chunk size and overlap, and their token lengths relative to the FEVER evidence sentences.
19. **State whether the corrupted-passage position is matched across strategies** (Section II.C) and give the selection seed; if unmatched, add position as a reported factor so that ordering effects are not confounded with strategy in Table III.
20. **Add a full-precision control run** of the same generator, to separate "small" from "quantized" in the abstention behavior reported in Section III.F (up to 69% for entity swap at 3/3) before attributing it to model scale in the Abstract and Section V.
21. **Remove or support the claim in Section IV that "raising k without raising source diversity offers little real protection."** $k = 3$ is fixed in all 588 runs (Section II.A, Section III.A), so the study contains no evidence about k . A sweep over k in $\{3, 5, 7\}$ at a fixed poison fraction would both support this claim and turn the "majority" pattern of Section III.E into a testable dose-response rather than an artifact of a 4-point grid.
22. **Implement at least one of the three defenses proposed in Section IV** – cross-passage disagreement flagging is the cheapest – and report its effect on poisoned accuracy and on false suppression of correct answers. As it stands, the paper's practical recommendations rest on no experiment.
23. **Extend Section III.H** with the adversarial-RAG literature (corpus poisoning, knowledge-base corruption, prompt injection). The current seven references contain no prior attack work, so the paper's novelty cannot be assessed.
24. **Tabulate the figure data.** Figures 4, 5, and 6 carry quantitative claims that appear nowhere in numeric form, including the 69% entity-swap abstention at 3/3 and the roughly 29% versus 20% overlap-proxy gap (Section III.F). Add the underlying values, with n per cell, in an appendix table.
25. **Explain the 12 duplicate queries** among the 49 (Section II.E) and either de-duplicate to 37 or justify the duplication, since it affects every denominator in Tables I-III and the clustering of the bootstrap in Section III.C.

5. Recommendation

Major revisions. The study is honestly framed, the limitations section anticipates several of the objections above, and – unusually – the reported statistics survive independent recomputation without a single arithmetic error, which speaks well of the author's care with the run log. But the paper does not yet

measure what its title claims: the retriever is never attacked, no indexed document is poisoned, and the threat model arrives in Section IV too late to constrain the design. Three of the four headline conclusions change or fall when the same table is recomputed – the accuracy drop should be 25.9 rather than 34.4 points, the strategy ranking of Table III reverses under a poison-adoption measure, the negation-resilience claim is an artifact of the keyword matcher, and the significance test in Section III.E flags the untreated 0/3 control at the same threshold as the attack. None of this is fatal: every item is addressable within the existing 588-run log plus one or two additional sweeps, and the underlying measurement is worth having. With the threat model repositioned and completed, the metrics repaired (items 6, 7, 9, 13), the inference corrected (items 10, 11, 12), and the artifacts released (item 17), this would be a useful small-scale reference point for practitioners running quantized RAG stacks. I would be glad to look at a revised version.

Declarations

Potential competing interests: No potential competing interests to declare.